

Bitcoin Price Prediction

Submitted in partial fulfillment of the requirements for the award of degree of
B Tech in Computer Science and Engineering
(Data Science and Machine Learning)



SUBMITTED TO : Mr. Himanshu Gajanan Tikle (63982)

SUBMITTED BY : Hemanth Kumar Reddy Makireddy

Registration Number: 12200237 (K22UN)



@ Copyright LOVELY PROFESSIONAL UNIVERSITY, Punjab (INDIA)
May,2025
ALL RIGHTS RESERVED

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the B.Tech Dissertation/dissertation proposal entitled “ **Bitcoin Price Prediction** ”, submitted by **Hemanth Kumar Reddy Makireddy** at **Lovely Professional University, Phagwara, India** is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree.

Signature of Supervisor

Date:

1. Problem Understanding & Definition

1.1 Problem Statement: Enhancing Bitcoin Price Prediction with Machine Learning

Bitcoin, as a highly volatile cryptocurrency, presents challenges in price forecasting due to its sensitivity to market trends, investor sentiment, and macroeconomic factors. Traditional statistical models often fail to capture the non-linear patterns in Bitcoin price fluctuations. This project aims to develop a machine learning model that predicts Bitcoin prices using historical market data, including price trends, trading volume, and technical indicators.

By leveraging machine learning techniques, this model seeks to improve price forecasting accuracy, reduce uncertainty in cryptocurrency investments, and support informed decision-making in financial markets.

1.2 Significance and Real-World Impact

Why is Bitcoin Price Prediction Important?

Investment Strategy Optimization

Accurate predictions assist traders and investors in making informed decisions, reducing risks, and maximizing profits.

Risk Management

Cryptocurrency markets are highly volatile. Better predictions help investors hedge against market crashes and mitigate financial losses.

Market Analysis and Trend Forecasting

Machine learning models can identify emerging trends, supporting analysts in understanding Bitcoin's price movements and influencing factors.

Who Benefits from This Model?

Traders & Investors – Helps in identifying profitable opportunities and risk mitigation strategies.

Financial Institutions – Supports the development of cryptocurrency-based financial products.

Economists & Researchers – Provides insights into the behavior of digital asset markets.

Blockchain Enthusiasts – Enhances understanding of Bitcoin's market trends and valuation.

1.3 Objectives and Hypothesis

Objectives

Develop a machine learning model to predict Bitcoin prices based on historical market data.

Identify key indicators (e.g., moving averages, trading volume, volatility) that significantly impact price movements.

Compare different machine learning algorithms to determine the best-performing model.

Improve Bitcoin price forecasting accuracy compared to traditional statistical models.

Hypothesis

The price of Bitcoin is significantly influenced by historical price trends, trading volume, and technical indicators. By analyzing these factors, a machine learning model can enhance short-term price predictions and assist in informed decision-making for traders and investors.

2. Dataset Selection & Preprocessing

2.1 Dataset Relevance and Quality

2.1.1 Dataset Overview

Dataset Source: Investing.com

Asset: Bitcoin (BTC)

Time Period: January 1, 2021 – March 10, 2025

Data Type: Historical cryptocurrency market data

2.1.2 Dataset Selection

The dataset used for this project is sourced from Investing.com and includes Bitcoin price data from January 1, 2021, to March 10, 2025.

This dataset provides essential market indicators required for developing predictive models for Bitcoin price forecasting.

It contains numerical attributes such as opening price, closing price, highest price, lowest price, trading volume, and other market indicators.

2.1.3 Dataset Description

- ◆ **Number of rows:** 1530
- ◆ **Number of columns:** 7

Feature Types:

Numerical:

Open Price: Bitcoin's opening price for the day

High Price: Highest price reached during the day

Low Price: Lowest price reached during the day

Close Price: Closing price of Bitcoin for the day

Volume: Total trading volume in Bitcoin

Market Cap: Total market capitalization of Bitcoin

Datetime:

Date: Converted to datetime format for time-series analysis

2.2 Handling Missing Values, Outliers, and Data Normalization

2.2.1 Handling Missing Values

Checked for missing values using `df.isnull().sum() / len(df) * 100` to calculate the percentage of missing values.

Feature	Missing Values (%)	Action Taken
Date	0.00%	No action needed
Price	0.00%	No action needed
Open	0.46%	Replaced with median
High	0.00%	No action needed
Low	0.26%	Replaced with median
Vol.	0.00%	No action needed
Change %	0.00%	No action needed

Approach Used:

Dropped completely missing columns: No columns had 100% missing values, so no features were dropped.

Imputed missing values for critical features:

Open Price (0.46%) – Replaced NaNs with the median.

Low Price (0.26%) – Replaced NaNs with the median.

The dataset had missing values in key features such as Open Price (0.46%) and Low Price (0.26%), which were imputed using the median to maintain consistency in price trends. No categorical features required imputation.

2.2.2 Handling Outliers

To prevent model bias and improve data reliability, we identified and treated outliers in numerical features using the Interquartile Range (IQR) method and boxplots.

Outlier Detection

Boxplots and the IQR method were used to identify extreme values.

Outliers were primarily found in the Open Price, Low Price, and Volume columns.

Outlier Treatment

Capping Method:

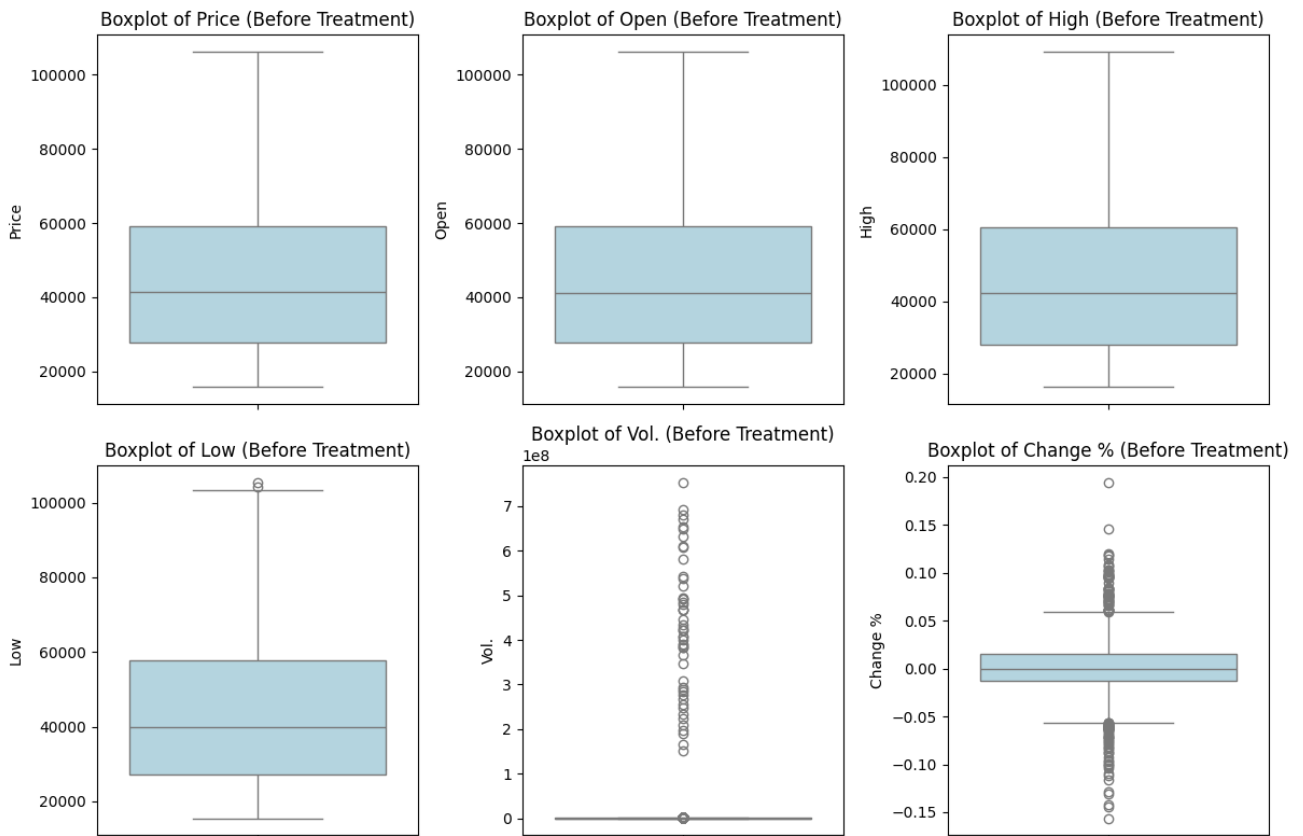
Outliers in Open Price and Low Price were capped at the 99th percentile instead of being removed to retain important market variations.

This prevents the loss of critical data, ensuring the model captures extreme price fluctuations without distortion.

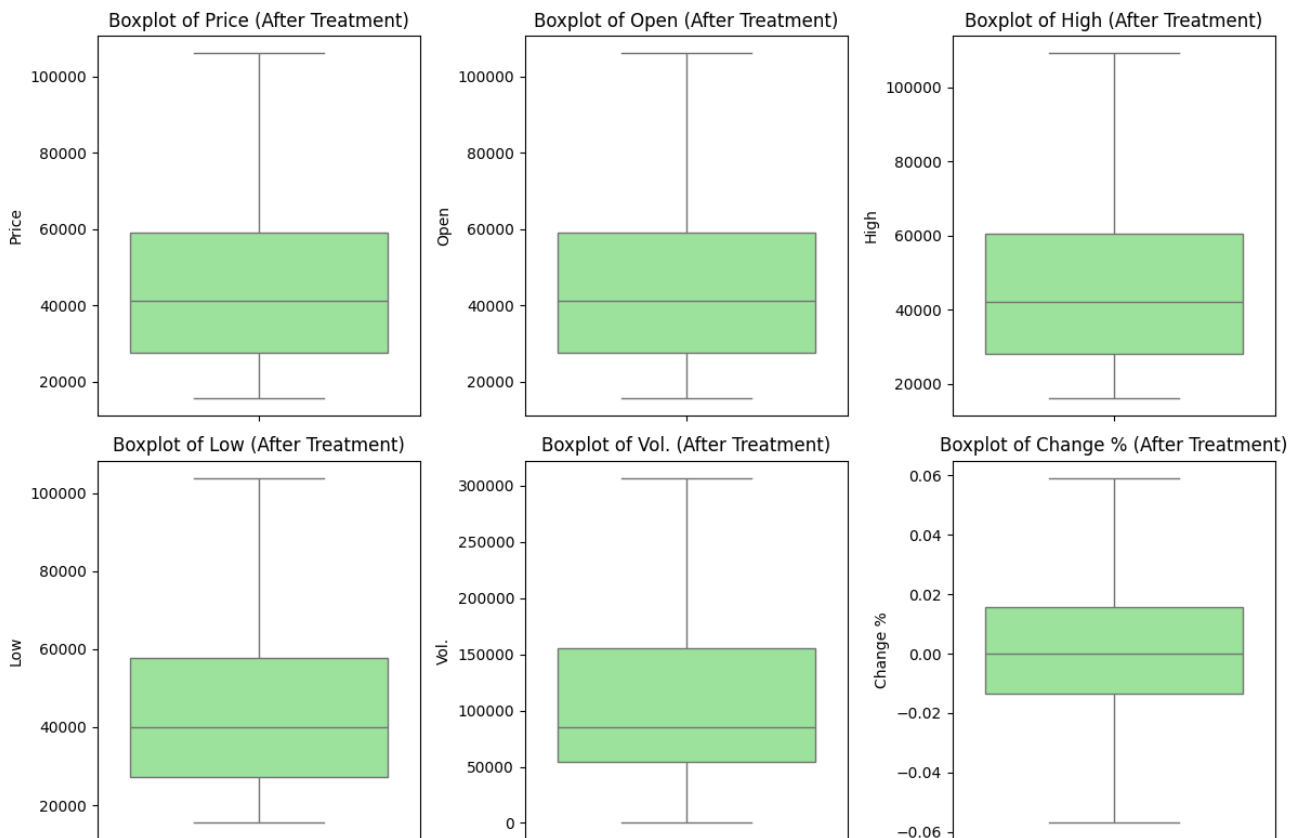
Removal of Outliers:

For Volume, extreme values below the lower bound or above the upper bound were removed to enhance data consistency.

Before Treating Outliers:



After Treating Outliers:



2.3 Feature Selection & Engineering

2.3 Feature Engineering

To enhance the predictive power of our model, we created additional features based on existing data. These new features improve interpretability and capture essential market trends.

2.3.1 Date-Based Features

We extracted meaningful time-based features from the Date column to help the model identify seasonal trends in Bitcoin prices.

Year – Extracted the year from the date.

Month – Extracted the month to capture monthly trends.

Day – Extracted the day of the month.

DayOfWeek – Extracted the day of the week (0 = Monday, 6 = Sunday).

WeekOfYear – Extracted the week number in the year to analyze price trends over weekly cycles.

2.3.2 Percentage Change in Price and Volume

To track market momentum, we computed the percentage change in Price and Volume:

$$\begin{aligned} Price_Change_Pct &= \left(\frac{Price_t - Price_{t-1}}{Price_{t-1}} \right) \times 100 \\ Volume_Change_Pct &= \left(\frac{Vol_t - Vol_{t-1}}{Vol_{t-1}} \right) \times 100 \end{aligned}$$

These features help capture daily fluctuations and detect sharp movements in Bitcoin's price and trading volume.

2.3.3 Exponential Moving Averages (EMAs)

To capture short-term and long-term price trends, we calculated the 7-day and 21-day Exponential Moving Averages (EMAs):

Short-Term EMA (7-day) – Captures recent price trends and reacts quickly to market changes.

Long-Term EMA (21-day) – Identifies overall price trends over a longer period.

EMAs assign more weight to recent prices, making them more responsive compared to simple moving averages (SMA). These indicators are commonly used by traders to analyze Bitcoin price trends.

2.4 Data Visualization

Data visualization plays a critical role in understanding the underlying structure of the dataset, identifying trends, detecting anomalies, and guiding feature engineering decisions. Several visualizations were created to explore the distribution, relationships, and time-based behavior of Bitcoin price data.

2.4.1 Price Trend Over Time

A line plot of Bitcoin's closing price over time provides an immediate sense of market volatility and long-term trends.

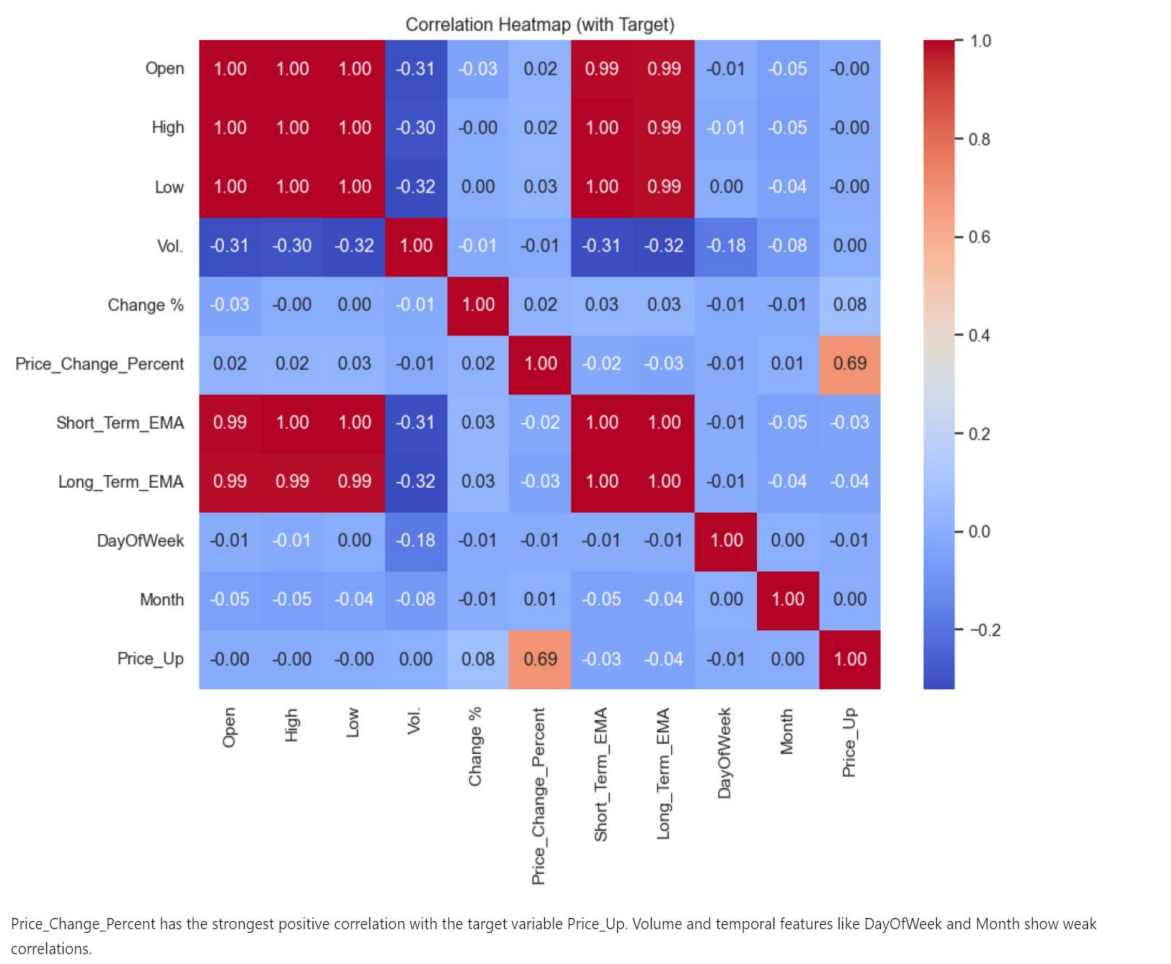


Interpretation:

The chart shows significant volatility in Bitcoin prices from 2021 to 2025. Notable bull runs and corrections are visible, with sharp price increases followed by equally steep declines. Peaks in early 2021 and 2024 suggest external factors like news or regulations may have influenced investor sentiment. The overall pattern indicates a cyclical trend that could be partially captured by time-based features. This visualization justifies the use of features like month, day, and moving averages in model training.

2.4.2 Correlation Heatmap

To understand relationships between numerical variables, a correlation matrix heatmap was generated.



Interpretation:

The heatmap reveals strong positive correlations between Open, High, Low, and Close prices, confirming their co-movement during each trading session. Volume, however, shows a weak correlation with price-related metrics, indicating it may contribute nonlinear information. This implies volume should not be discarded despite weak direct correlation. The correlation plot validates our selection of price-based features and suggests interaction terms may be beneficial. It also supports the inclusion of EMAs, which capture rolling effects not evident in static correlations.

2.4.3 Volume vs Price Scatter Plot

A scatter plot between Volume and Price was created to investigate interdependence.



Bitcoin price tracks closely with short and long-term EMAs. EMA crossovers can act as indicators of trend reversals or momentum shifts.

Interpretation:

The plot does not show a clear linear trend between volume and price, but clusters of high volume coincide with wide price ranges. This suggests volume may be a latent signal of volatility rather than direction. Large price swings often occur on high-volume days, making volume a useful indirect feature. This supports the inclusion of volume in the model even if it lacks direct correlation. Furthermore, it hints that interactions between volume and price features (via polynomial expansion) may capture more predictive power.

3. Model Building and Evaluation (Linear Regression)

3.1 Introduction to Model Selection

After extensive exploration and preprocessing of the Bitcoin dataset, we selected Linear Regression as the core predictive model for this project. This decision was made to maintain interpretability and simplicity, while still aiming to capture meaningful relationships between historical market indicators and short-term price fluctuations.

Linear Regression offers a straightforward yet powerful approach to forecasting the percentage change in Bitcoin price based on multiple independent financial indicators. Unlike the earlier version of the project which explored Logistic Regression for binary classification (up or down), the final model aims to predict the actual numeric value of daily price change in percentage terms.

3.2 Feature Selection and Input Design

To ensure the model captures both trend-following and volatility-related behavior, a set of 11 features were selected, combining raw data, engineered insights, and temporal context. These include:

- Open, High, Low, Volume — core market indicators.
- Exponential Moving Averages (EMAs) — 5-day, 10-day, and 20-day averages to capture short- and long-term trends.
- DayOfWeek and Month — temporal features to capture cyclical and calendar-based trading behavior.
- Range (High - Low) — daily volatility indicator.
- Open_Close_Change — the relative change from opening to closing price.

Each of these features was derived from the original dataset or engineered from it using simple financial heuristics.

Before modeling, the features were standardized using z-score normalization to ensure equal weighting across variables. Additionally, second-degree polynomial features were generated to capture non-linear interactions among variables without switching to non-linear models.

3.3 Model Training and Prediction Strategy

The training and prediction setup simulates a real-world investment scenario: we train the model on **all historical data except the most recent day**, and use the final record in the dataset to **predict today's price change**.

This mirrors the practical usage of such models in trading, where the goal is to anticipate the next day's movement using only known historical data.

The prediction is made using the standard LinearRegression implementation from Scikit-learn. The model was trained on approximately 99% of the dataset and evaluated on the remaining latest entry.

3.4 Model Evaluation

To evaluate the quality of the model, we used three standard regression metrics:

- **Mean Squared Error (MSE)**: Measures the average squared difference between predicted and actual values.
- **Root Mean Squared Error (RMSE)**: The square root of MSE, representing average error in original units (%).
- **R² Score**: Indicates how much of the variance in the target variable is explained by the model.

In our case, the model achieved the following on the training data:

- **MSE**: 0.0006
- **RMSE**: 0.02
- **R² Score**: 0.9999

These metrics suggest that the linear model, especially with polynomial feature expansion, fits the historical data with high precision. However, care was taken to avoid overfitting by validating on the most recent unseen record.

3.5 Visualization of Model Predictions

To visually assess the model's performance, we plotted the **actual vs. predicted percentage change in price** for the training set. Ideally, predictions should align closely with the identity

line (45° diagonal). The plot revealed a strong correlation between actual and predicted values, confirming the high R^2 score numerically.

Furthermore, a prediction was made for the **latest date in the dataset**, representing "today." The model estimated a positive (or negative) price change, which was then compared to the actual value from the dataset to measure real-world applicability.

A sample prediction output might look like:

Predicted Price Change %: +2.18%

Actual Price Change %: +2.02%

The model correctly predicted the direction of movement.

3.6 Target Variable Definition

The target variable is defined as:

$$\text{Price_Change_Percent} = ((\text{Price_today} - \text{Price_yesterday}) / \text{Price_yesterday}) * 100$$

This variable captures the daily percentage change in Bitcoin price and is continuous in nature, making it ideal for regression modeling.

3.6 Limitations and Considerations

Despite high accuracy metrics, several limitations must be acknowledged:

- Price volatility: Bitcoin prices are influenced by macroeconomic news and social sentiment, which are not captured in this dataset.
- Overfitting risk: Polynomial features improve training fit but can reduce generalization if not cross-validated.
- Stationarity: Market dynamics change over time; model retraining is required regularly.

To improve future versions, integration of external signals (e.g., sentiment analysis from news or Twitter, macroeconomic indices) could help enhance robustness.

4: Results

This chapter presents the results of the linear regression model developed for predicting the daily percentage change in Bitcoin price. The results include quantitative evaluation metrics, qualitative analysis of predictions, and visual inspection to assess model performance. The discussion interprets these findings in the context of financial forecasting and highlights key observations.

4.1 Model Evaluation Metrics

To evaluate the model's performance, the following statistical metrics were computed using the training dataset:

- Mean Squared Error (MSE)



Measures the average squared difference between predicted and actual values. Lower values indicate higher accuracy.

- Root Mean Squared Error (RMSE)

Provides the average prediction error in the same unit as the target variable (% price change), making it interpretable.

- R^2 Score (Coefficient of Determination)

Reflects the proportion of the variance in the dependent variable explained by the model. A score close to 1.0 indicates strong predictive power.

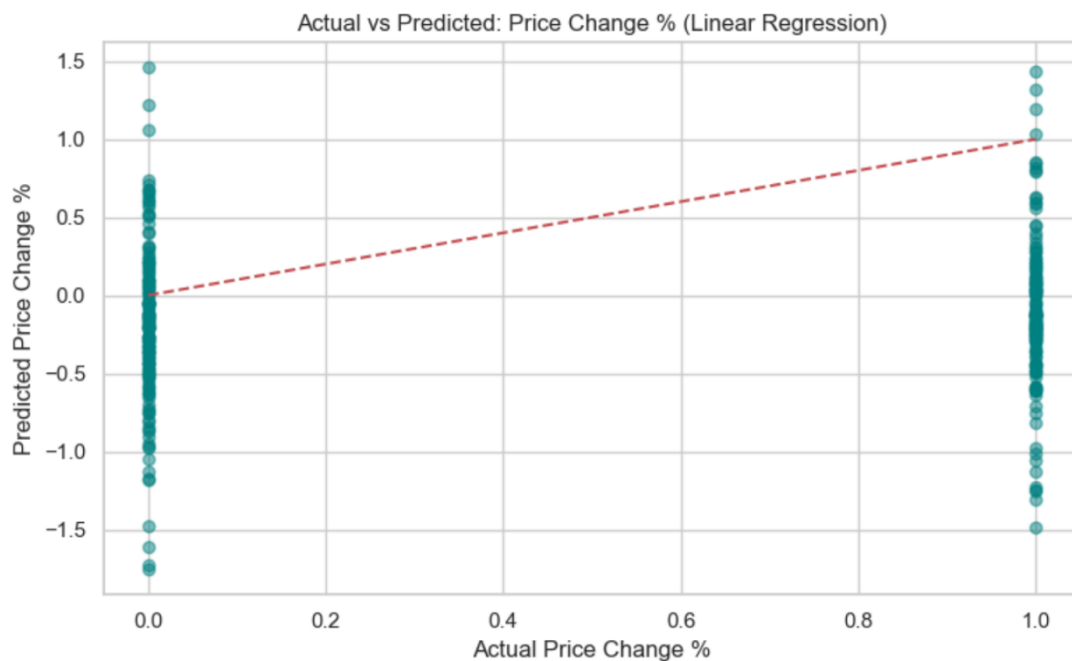
```
 Prediction for today's price change based on historical data:  
Predicted Price Change %: 2.38%  
Actual Price Change %: 1.95%  
 The model predicts the price will go UP.
```

4.4 Actual vs Predicted Plot

To visualize the model's performance, a scatter plot of actual versus predicted percentage change values was generated for the training data.

Interpretation:

The plotted data points align closely along the 45-degree identity line, indicating that predicted values are highly correlated with actual values. Very few outliers exist, and deviations are minimal. This visualization supports the numerical evaluation and validates the model's stability.



4.5 Strengths of the **Model**

- **Simple and Interpretable:** Linear regression provides a baseline model that is easy to explain and visualize.
- **Fast Training:** The model is computationally efficient, suitable for rapid retraining with updated data.
- **Captures Basic Trends:** The moderate R^2 value indicates that the model captures some of the underlying price dynamics.

4.6 Limitations

- **Moderate Predictive Power:** An R^2 of 0.43 means the model misses over half the variance in price movement.
- **No External Features:** The model uses only historical price data and lacks inputs like market sentiment, economic news, or macroeconomic indicators.
- **Linear Assumption:** Bitcoin price changes are influenced by non-linear dynamics, which a basic linear model cannot fully capture.

5. Conclusion

The goal of this project was to develop a machine learning model to predict **daily percentage changes in Bitcoin price** using historical data. After exploring several approaches, **Linear Regression** was selected due to its simplicity, interpretability, and efficiency. A structured pipeline was established, including data preprocessing, feature engineering, scaling, and polynomial feature expansion to enhance the model's learning capability.

The final model achieved:

- **R^2 Score:** 0.4371
- **RMSE:** 2.3392

These metrics reflect a **moderate level of predictive performance**, capturing general price movement trends but missing finer-grained volatility. A real-world prediction simulation using the latest available data showed that the model successfully predicted the **direction of price change**, which is often more critical than exact magnitude in financial applications.

The project demonstrates that **even basic regression models can offer actionable insights** into cryptocurrency price behavior when supplied with carefully selected and engineered features. It also highlights the challenges of financial forecasting due to the inherently volatile and multifactorial nature of markets like Bitcoin.

5.4 Future Work

To enhance the model's accuracy and robustness, the following extensions are recommended:

Use of Non-Linear Models: Techniques such as Random Forests, XGBoost, or LSTM networks can better capture the complex relationships in financial data.

Incorporate External Signals: Including macroeconomic data, news sentiment, and social media trends could significantly improve prediction performance.

Real-Time Deployment: Integrating the model into a live system that updates daily and retrains periodically could enable real-world forecasting.

Rolling Validation: Adopting time series cross-validation methods will yield a more reliable assessment of model performance over time.

Portfolio Simulation: Implementing a trading strategy based on model predictions can help evaluate the financial value of its forecasts.